(1)

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GA/AA/88D-13

ESTIMATED SATELLITE CLUSTER ELEMENTS IN NEAR CIRCULAR ORBIT

THESIS

Michael L. P. Ward
Captain, USAF

AFIT/GA/AA/88D-13

Approved for public release; distribution unlimited

AFIT/GA/AA/88D-13

ESTIMATED SATELLITE CLUSTER ELEMENTS IN NEAR CIRCULAR ORBIT

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Astronautical Engineering

Michael L. P. Ward, B.S.
Captain, USAF

December, 1988

Preface

This study investigated the performance of an on-board filter algorithm in estimating the relative positions of each element in a ten satellite orbital cluster. The estimator chosen was the U-D covariance factorization version of the Kalman filter.

The performance was evaluated by comparing the square-root of the position covariance eigenvalues to the magnitude of the true position errors. Even though filter tuning was limited to the diagonals of the dynamics noise matrix, the position errors were well within the accuracy requirements for all test cases.

In programing and writing my thesis, I received invaluable help from my faculty advisor, Dr. William Wiesel. His patience and guidance are greatly appreciated. I also thank Dr. Peter Maybeck for his counsel. Special thanks to my wife Susan and son David for their understanding and support over the past months. Finally, I wish to thank the Lord who gave me the strength to keep going during the long hours.

Michael L. P. Ward

Table of Contents

List of Figures

List of Tables

AFIT/GA/AA/88D-13

Abstract

The relative position determination of the satellites in an orbital cluster is investigated. The on-board estimator is the U-D covariance factorisation filter with dynamics based on the Clohessy-Wiltshire equations. Performance is measured by comparing the square root of the position covariance eigenvalues to the magnitude of true position errors. True errors are also compared to minimum accuracy requirements. Filter tuning is limited to adjusting the diagonal entries of the dynamics noise matrix. Test cases include: perfect initial conditions with perfect range data; and perfect initial conditions with noise corrupted measurements.

ESTIMATED SATELLITE CLUSTER ELEMENTS IN NEAR CIRCULAR ORBIT

I. Introduction

The arraying of satellites into clusters has been under consideration for a number of years for commercial communications applications. Stationing these satellites in a restricted section of an orbit allows for in-orbit spares to replace satellites as they fail, thereby dramatically increasing overall system reliability. Reliability is, of course, of major concern for military applications as well. An additional benefit to the military planner is increased wartime system survivability. A cluster basing strategy allows for the use of less expensive, less sophisticated satellites. Survivability is enhanced because the loss of a satellite does not destroy the system.

One proposed use of such a satellite cluster would be a space based radar. In order to form a cohesive image of the radar return, the relative positions of the satellites need to be known to within one quarter of the radar wavelength [3:page 443]. For the purposes of this thesis, 25 meters will be considered sufficiently accurate (see appendix A).

A number of authors have published studies in the area of satellite clusters. The idea was first introduced by P. S. Visher, who was granted a patent in 1983 [2:page 1713]. Many authors have done follow-up studies on cluster geometries and station keeping [6,2]. Few have addressed the stochastic nature of the position determination problem.

One writer who has addressed this problem is John Murdoch [8]. Specifically, he examined the possibility of on-board relative position determination of

the satellites. However, he performed a least-squares analysis which is less appropriate than a recursive filter for on-board applications. Also, his cluster consisted of only two satellites in near-geostationary orbits. An actual cluster would very likely contain many more satellites.

This thesis is not concerned with a specific cluster geometry nor with station keeping but with the study of a recursive filter for on-board relative position determination. The cluster consists of up to ten satellites in near-circular, low-earth orbit. To complete the study, a truth model based on the two-body equations of motion was developed to provide measurement data to a simulated on-board Kalman Filter. A single sample Monte Carlo analysis was performed to rate filter performance by comparing the covariance standard deviation to the true error.

II. Problem Description

2.1 Introduction

In the satellite cluster concept, each satellite would estimate the positions of every element of the array to get its own position. A pulse used to synchronise the on-board clocks of the satellites could be used to provide range data z for the estimator. For the remainder of this study, the filter algorithm will be designed for satellite one but can be generalised for the other satellites. Figure 2.1 illustrates the basic concept and will be referred to in later analysis.

In order to test the concept of an on-board filter, it was necessary to build a real world or *truth model* against which the filter performance could be judged. Since the purpose of this thesis was to evaluate the concept, the truth model was simplified to the two-body equations of motion. The truth model outputs the *true state* $\mathbf{x}_t(t)$ and the *measurements* $\mathbf{z}_t(t)$. These measurements are used by the satellite Kalman filter (figure 2.2) to output an *estimated state* $\hat{\mathbf{x}}(t)$. The critical elements of the filter states $\hat{\mathbf{y}}(t)$ are then compared to the corresponding elements of the true state $\mathbf{y}(t)$ after measurement update to arrive at the *true error* $\mathbf{e}_t(t)$.

$$\mathbf{e}_t = \hat{\mathbf{y}} - \mathbf{y}_t \tag{2.1}$$

This true error can then be compared to the square root of the appropriate eigenvalues of the *covariance matrix* $\mathbf{P}$ to see if the filter performs as well as it believes it is performing [4:page 339].

2.2 Truth Model

The truth model is based on a group of up to ten satellites orbiting the earth as a single unit. In order to maintain cluster integrity, they have to remain within a prescribed spatial volume. This means all satellite orbital periods must be equal

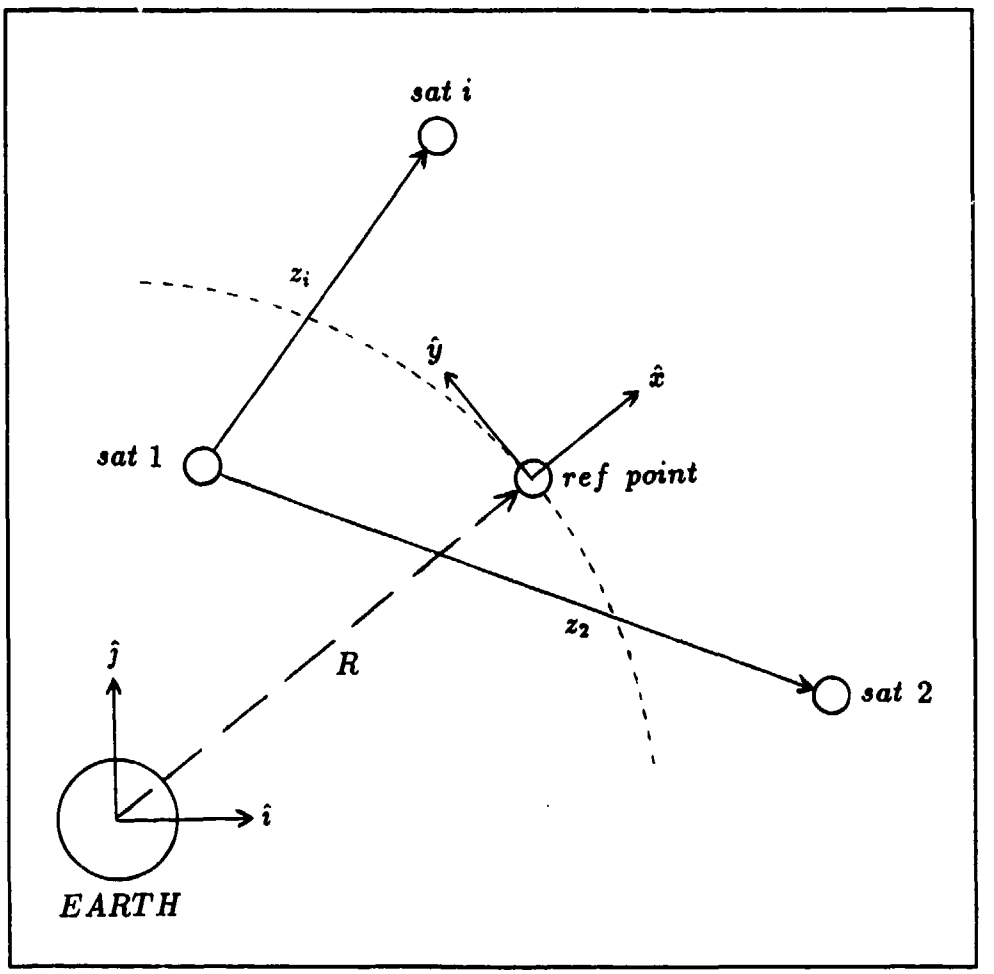

Figure 2.1. Two-Body Relative Motion

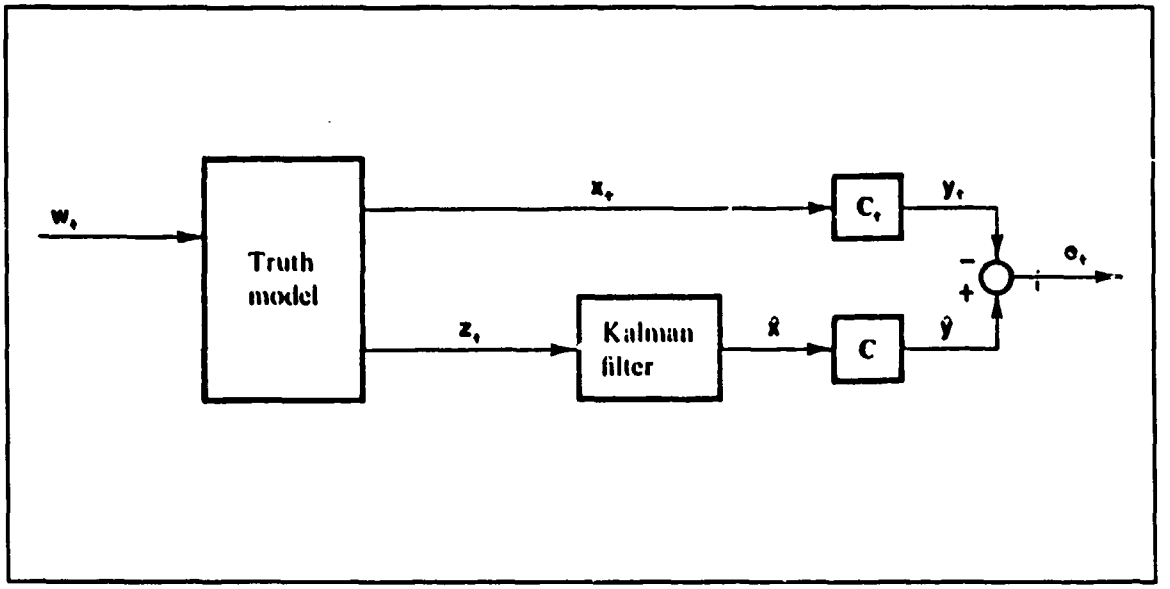

Figure 2.2. Performance Evaluation of a Kalman Filter

or the formation will drift apart. Therefore, the model first creates the cluster and then propagates each satellite through its orbit.

As stated earlier, the output of the truth model is relative range data, for input to the Kalman filter, and true position, for comparison to the estimated position. Since the measurement data is relative information, the absolute position cannot be estimated. Instead, position is described with respect to a rotating reference point as shown in Figure 2.1

This point is the origin of a reference frame $(\hat{x},\hat{y},\hat{z})$ rotating with respect to an earth centered inertial frame $(\hat{\imath},\hat{\jmath},\hat{k})$ at a constant *angular rate* $\omega\,\hat{k}$. Note $(\hat{\imath},\hat{\jmath})$ and $(\hat{x},\hat{y})$ are always coplanar and that $\hat{k}$ is always parallel to $\hat{z}$. The reference point is in a circular orbit at radius R and at circular orbit velocity of

$$v_{co} = \sqrt{\frac{\mu}{R}}$$

where μ is the gravitational parameter of the earth. The initial position of the

reference point in the fixed frame is

$$\mathbf{r}_{ref} = \begin{bmatrix} R & 0 & 0 \end{bmatrix}^T \tag{2.2}$$

with an initial velocity of

$$\mathbf{v}_{ref} = \begin{bmatrix} 0 & v_{cs} & 0 \end{bmatrix}^T \tag{2.3}$$

Since this thesis is not concerned with a specific geometry, the satellites of the cluster are positioned at random about the reference point but within a specified radius *rad*:

$$\mathbf{r}_i = \begin{bmatrix} R \\ 0 \\ 0 \end{bmatrix} + \mathbf{n}_i \, rad \tag{2.4}$$

where $i = 1, 2, \ldots, s$ (the total number of satellites in the cluster) and where $\mathbf{n}_i$ is a particular vector realization of a uniform random number generator with outputs between -0.5 and 0.5. Appropriate values for the velocity vector $\mathbf{v}_i$ can be found from the Clohessy-Wiltshire equations [9] (these equations will be explained in detail in the next section):

$$\begin{aligned} v_i \, \hat{\imath} &= \eta(r_i \, \hat{\imath} - r_{ref} \, \hat{\imath}) \\ v_i \, \hat{k} &= \eta r_i \, \hat{k} \end{aligned} \tag{2.5}$$

where η is the mean motion. As previously stated, the velocity must be specified so that the period of all satellites are equal. Since the period depends only on the size of the *semi-major axis* a [1:page 33], this criterion can be used:

$$\frac{\mathbf{v}_i^2}{2} - \frac{\mu}{\mathbf{r}_i} = \frac{-\mu}{2a} \tag{2.6}$$

Solving for $v_i \, \hat{\jmath}$ yields

$$v_i \, \hat{\jmath} = (2\mu(\frac{1}{\mathbf{r}_i} - \frac{1}{2a}) - v_i^2 \, \hat{\imath} - v_i^2 \, \hat{k})^{\frac{1}{2}} \tag{2.7}$$

with the positive square root chosen so that each satellite will maintain formation integrity.

By using this position and velocity information, the input for the Kalman filter can be formed. Let $\mathbf{x}_t(t=0)$ be the initial true state defined as the position and velocity of each satellite in the rotating frame (ROT). Initially, the axes of both coordinate frames are aligned therefore the position of each satellite in the rotating frame is

$$\mathbf{r}_{i[ROT]} = (\mathbf{r}_i - \mathbf{r}_{ref})_{[FIX]} \tag{2.8}$$

and the velocity is

$$\mathbf{v}_{i[ROT]} = \mathbf{v}_{i[FIX]} - \omega \times \mathbf{r}_{i[FIX]} \tag{2.9}$$

where $\omega = v_{cs}/R$. The velocity can also be expressed in terms of relative position as

$$\mathbf{v}_{i[ROT]} = (\mathbf{v}_i - \mathbf{v}_{ref})_{[FIX]} - \omega \times \mathbf{r}_{i[ROT]} \tag{2.10}$$

therefore the state at $t = 0$ is

$$\mathbf{x}_t(0) = \begin{bmatrix} \mathbf{r}_{1[ROT]} \\ \mathbf{v}_{1[ROT]} \\ \vdots \\ \mathbf{r}_{s[ROT]} \\ \mathbf{v}_{s[ROT]} \end{bmatrix} \tag{2.11}$$

The other desired output is a $s-1$ dimensional measurement vector of the form:

$$\mathbf{z}_t = \begin{bmatrix} |\mathbf{r}_1 - \mathbf{r}_2| \\ \vdots \\ |\mathbf{r}_1 - \mathbf{r}_s| \end{bmatrix} + \mathbf{u}_t \tag{2.12}$$

where $\mathbf{u}_t$ is a zero-mean white Gaussian noise with covariance $\mathbf{R}_t$ [4:page 330]. This noise is the best guess of the real world errors in the measurement process. The range measurements are computed from impulse signals sent to each satellite by every other cluster element at specified times. The major errors result from clock synchronization differences between satellites as well as the resolution limitations of this on-board clock [8:page 1016].

For this study, noise sources are not specified individually but are lumped together into a total generic measurement error *err*. This *err* is multiplied by the output of a Gaussian random number generator with statistics of mean of 0 and standard deviation of ± 1.0. Thus, $\mathbf{R}_t$ is an $(s-1)$-by-$(s-1)$ matrix with diagonal entries of err^2 and off-diagonal entries of zero. Note the off-diagonal zeros imply that the measurements are independent; this assumption is unrealistic given this problem statement but adequate for a proof-of-concept study. Some other possible error sources are biases caused by misalignments in tracking sensors. These biases are not addressed in this thesis.

The next task of the truth model is determining the future positions and velocities of the satellites. This requires solving the Kepler problem using the *f and g equations* in terms of *eccentric anomaly* E [1:page 219]. This solution begins with determining additional initial conditions.

Assume the initial position $\mathbf{r}_0$ and velocity $\mathbf{v}_0$ of each satellite is known in the inertial frame. Then the initial eccentric anomaly E_0 and *mean anomaly* M_0 for each satellite can be found by the following procedure:

$$\epsilon = \frac{1}{\mu}\left[\left(v_0^2 - \frac{\mu}{r_0}\right)\mathbf{r}_0 - (\mathbf{r}_0 \bullet \mathbf{v}_0)\,\mathbf{v}_0\right] \tag{2.13}$$

where ϵ is the *eccentricity vector* (constant for each satellite)

$$\nu_0 = \cos^{-1}\left(\frac{\epsilon \bullet \mathbf{r}_0}{\epsilon\, r_0}\right) \tag{2.14}$$

where ν is the *true anomaly*

$$E_0 = \cos^{-1}\left(\frac{\epsilon + \cos\nu_0}{1 + \epsilon\cos\nu_0}\right) \tag{2.15}$$

and finally

$$M_0 = E_0 - \epsilon\sin E_0 \tag{2.16}$$

With these initial conditions in hand, the Kepler problem can now be solved. The Kepler problem is simply the solution to the equation [1:page 220]

$$M = E - \epsilon\sin E$$

at some future time t. M at this time is easily found by

$$M = \sqrt{\frac{\mu}{a^3}}\,t - 2k\pi + M_0 \tag{2.17}$$

where k is an integer number equal to the number of times the satellite has passed through M_0. Since E cannot be solved for directly, it must be determined through an iteration scheme. The technique chosen for this thesis was a Newton iteration method. Starting with a first guess of $E_n = M$, the corresponding value of M_n using Eq (2.16) is

$$M_n = E_n - \epsilon\sin E_n$$

resulting in a new guess (E_{n+1}) of

$$E_{n+1} = E_n + \frac{M - M_n}{1 - \epsilon\cos E_n} \tag{2.18}$$

This iteration continues until the difference $M - M_n$ becomes sufficiently small and thus determines the final value of the eccentric anomaly E_f.

Now, the $\mathbf{r}$ and $\mathbf{v}$ vectors of each satellite can be found [1:page 198]:

$$\mathbf{r} = f\,\mathbf{r}_0 + g\,\mathbf{v}_0 \tag{2.19}$$

$$\mathbf{v} = \dot{f}\,\mathbf{r}_0 + \dot{g}\,\mathbf{v}_0 \tag{2.20}$$

where $f, g, \dot{f}, \dot{g}$ are

$$f = 1 - \frac{a}{r_0}(1 - \cos \Delta E) \tag{2.21}$$

$$g = t - \sqrt{\frac{a^3}{\mu}}(\Delta E - \sin \Delta E) \tag{2.22}$$

$$\dot{f} = -\frac{\sqrt{\mu\, a}\sin \Delta E}{r\, r_0} \tag{2.23}$$

$$\dot{g} = 1 - \frac{a}{r}(1 - \cos \Delta E) \tag{2.24}$$

The two unknown values in the above equations (ΔE and r) are determined by

$$\Delta E = E_f - E_0 \tag{2.25}$$

$$r = a(1 - \epsilon \cos E_f) \tag{2.26}$$

Since r and v of each satellite are now known, the relative position to the reference point in the rotating frame can be found. First, the current location of the reference point in the inertial frame can readily be determined by:

$$\mathbf{r}_{ref[FIX]} = \begin{bmatrix} R\cos\theta \\ R\sin\theta \\ 0 \end{bmatrix} \tag{2.27}$$

$$\mathbf{v}_{ref_{(FIX]}} = \begin{bmatrix} -v_{cs}\sin\theta \\ v_{cs}\cos\theta \\ 0 \end{bmatrix} \tag{2.28}$$

where $\theta = \omega\, t$. The relative position and velocity is then

$$\mathbf{r}_{rel[FIX]} = [\mathbf{r}(t) - \mathbf{r}_{ref}(t)] = r_1\hat{\imath} + r_2\hat{\jmath} + r_3\hat{k} \tag{2.29}$$

$$\mathbf{v}_{rel[FIX]} = [\mathbf{v}(t) - \mathbf{v}_{ref}(t)] = v_1\hat{\imath} + v_2\hat{\jmath} + v_3\hat{k} \tag{2.30}$$

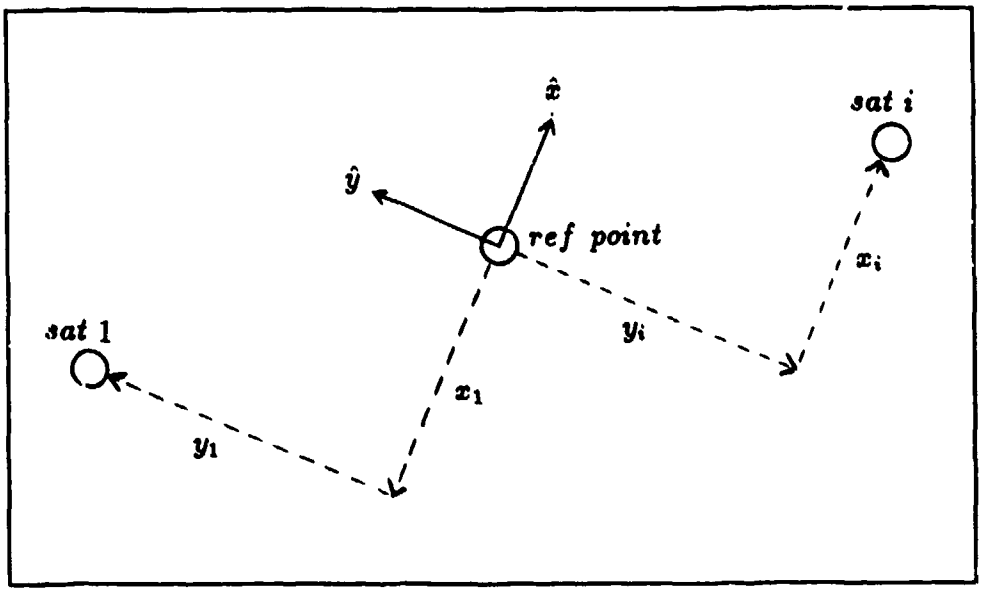

Figure 2.3. Cluster State

Unlike the situation in Eqs 2.8 and 2.10, the inertial and rotating coordinate frames are not aligned. Therefore, a coordinate transformation must be performed resulting in the following equation:

$$\begin{bmatrix} \mathbf{r}_{[ROT]} \\ \mathbf{v}_{[ROT]} \end{bmatrix} = \begin{bmatrix} r_1 \cos\theta + r_2 \sin\theta \\ -r_1 \sin\theta + r_2 \cos\theta \\ r_3 \\ v_1 \cos\theta + v_2 \sin\theta + \omega r_2 \\ -v_1 \sin\theta + v_2 \cos\theta - \omega r_1 \\ v_3 \end{bmatrix} \equiv \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{2.31}$$

The true state vector is formed according to Eq 2.11 and illustrated in Figure 2.3. The out of plane position z is not shown but is measured in the same way as the illustrated x and y components. Finally, the measurement vector is formed according to Eq 2.12.

2.3 *Kalman Filter*

Now that the 'real world' model has been completed to provide the necessary data for the filter, the filter itself can be developed. Assume the system description is in the form of a linear stochastic differential equation (to be developed later) which describes state propagation; with discrete-time, noise corrupted, nonlinear measurements as available inputs. The appropriate estimator is the *extended Kalman Filter*. Extended, because the nonlinear measurements are linearized about the new state estimate at each sample time as opposed to being linearized about a fixed nominal state for all time [5:page 42].

Since the dynamics are linear, the standard Kalman filter propagation equations can be used [4:page 220].

$$\hat{\mathbf{x}}(t_i^-) = \Phi(t_i, t_{i-1})\hat{\mathbf{x}}(t_{i-1}^+) \tag{2.32}$$

$$\mathbf{P}(t_i^-) = \Phi(t_i, t_{i-1})\mathbf{P}(t_{i-1}^+)\Phi^T(t_i, t_{i-1}) + \mathbf{G}_d(t_{i-1})\mathbf{Q}_d(t_{i-1})\mathbf{G}_d^T(t_{i-1}) \tag{2.33}$$

where

$\mathbf{Q}_d$ The covariance of the dynamics driving noise.

$\hat{\mathbf{x}}(t_i^-)$ The estimated state after time propagation.

$\hat{\mathbf{x}}(t_i^+)$ The estimated state after the measurement update.

$\mathbf{G}_d$ Equals the identity matrix (because the model is an equivalent discrete-time representation of a continuous-time system [4:page 377]).

Φ The state transition matrix.

For convenience, the time argument will be dropped for the remainder of this discussion and $(t_i^{\pm})$ will be written as $(\pm)$. The first time propagation occurs prior to any measurement updates, therefore the initial conditions $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$ must be determined. As a first cut, let $\hat{\mathbf{x}}_0$ equal $\mathbf{x}_t(0)$ from the truth model with the position measured in kilometers and velocity in kilometers per second. Also, let $\mathbf{P}_0$ be diagonal with position and velocity covariances of order $10^{-2}km^2$ and

$10^{-8}(km/sec)^2$ respectively. These values result from an assumed initial position accuracy of $10\,m$ and a corresponding velocity error of $2\,m/sec$. This $\mathbf{P}_0$ can be replaced with the steady-state value once the filter has been tested. Let $\mathbf{Q}_d$ equal the squared difference between $\mathbf{x}_t$ and $\hat{\mathbf{x}}$ after the first time propagation. A sample time of 350 seconds was chosen to ensure the relative satellite motion could be observed.

Because of the nonlinearity of the measurements, the extended Kalman filter update equations must be used [5:page 44].

$$\mathbf{K} = \mathbf{P}(-)\mathbf{H}^T\{\,\mathbf{H}\mathbf{P}(-)\mathbf{H}^T + \mathbf{R}_f\,\}^{-1} \tag{2.34}$$

$$\hat{\mathbf{x}}(+) = \hat{\mathbf{x}}(-) + \mathbf{K}\,\{\,\mathbf{z} - \mathbf{h}[\hat{\mathbf{x}}(-)]\,\} \tag{2.35}$$

$$\mathbf{P}(+) = \mathbf{P}(-) - \mathbf{K}\mathbf{H}\mathbf{P}(-) \tag{2.36}$$

$\mathbf{K}$ is the Kalman filter gain and $\mathbf{R}_f$ is the filter measurement noise covariance equaling $\mathbf{R}_t$ from the truth model.

For an understanding of the h vector, refer back to Eq 2.12. The expectation of this equation can be written as

$$\hat{\mathbf{z}} = \mathbf{h}(\,\hat{\mathbf{x}}(t)\,)$$

Therefore, $\mathbf{h}(\hat{\mathbf{x}}, t)$ is the filter's estimate of range and

$$\mathbf{z} - \mathbf{h}[\hat{\mathbf{x}}(-)]$$

is the residual or true error in range used to update the state. Thus, h has the form

$$\mathbf{h} = \begin{bmatrix} \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \\ \vdots \\ \sqrt{(x_1 - x_s)^2 + (y_1 - y_s)^2 + (z_1 - z_s)^2} \end{bmatrix} = \begin{bmatrix} h_1 \\ \vdots \\ h_{s-1} \end{bmatrix} \tag{2.37}$$

The matrix $\mathbf{H}$ comes from linearizing the h vector and evaluating it at $\hat{\mathbf{x}}(-)$:

$$\mathbf{H} \triangleq \left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(-)} \tag{2.38}$$

Which has the form:

$$\mathbf{H} = \begin{bmatrix} H_1 & -H_1 & 0 & \cdots & 0 \\ H_2 & 0 & -H_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ H_{s-1} & 0 & 0 & \cdots & -H_{s-1} \end{bmatrix} \tag{2.39}$$

where

$$H_1 = \begin{bmatrix} \frac{x_1-x_2}{h_1} & \frac{y_1-y_2}{h_1} & \frac{z_1-z_2}{h_1} & 0 & 0 & 0 \end{bmatrix} \tag{2.40}$$

$$H_{s-1} = \begin{bmatrix} \frac{x_1-x_s}{h_{s-1}} & \frac{y_1-y_s}{h_{s-1}} & \frac{z_1-z_s}{h_{s-1}} & 0 & 0 & 0 \end{bmatrix} \tag{2.41}$$

2.3.1 First Filter. The first filter designed and tested was the extended Kalman filter described above. The only element in the algorithm that has not been fully explained in this text is the Φ matrix. The state transition matrix was developed from the Clohessy-Wiltshire equations of motion [9:page 3] as

$$\ddot{x} - 2\eta\dot{y} - 3\eta^2 x = 0 \tag{2.42}$$

$$\ddot{y} + 2\eta\dot{x} = 0 \tag{2.43}$$

$$\ddot{z} + \eta^2 z = 0 \tag{2.44}$$

These equations can be integrated about the initial conditions $x_0, \dot{x}_0, y_0$, etc. at $t = 0$ to obtain the position and velocity solutions [9:pages 3-5]

$$x(t) = -\left(\frac{2}{\eta}\dot{y}_0 + 3x_0\right)\cos\eta t + \frac{\dot{x}_0}{\eta}\sin\eta t + 4x_0 + \frac{2}{\eta}\dot{y}_0 \tag{2.45}$$

$$y(t) = y_0 - (3\dot{y}_0 + 6\eta x_0)\,t + \left(\frac{4\dot{y}_0}{\eta} + 6x_0\right)\sin\eta t + \frac{2\dot{x}_0}{\eta}\cos\eta t - \frac{2\dot{x}_0}{\eta} \tag{2.46}$$

$$x(t) = x_0\cos\eta t + \frac{\dot{x}_0}{\eta}\sin\eta t \tag{2.47}$$

$$\dot{x}(t) = (2\dot{y}_0 + 3\eta x_0)\sin\eta t + \dot{x}_0\cos\eta t \tag{2.48}$$

$$\dot{y}(t) = -3\dot{y}_0 - 6\eta x_0 + (6\eta x_0 + 4\dot{y}_0)\cos\eta t - 2\dot{x}_0\sin\eta t \tag{2.49}$$

$$\dot{x}(t) = -x_0\eta\sin\eta t + \dot{x}_0\cos\eta t \tag{2.50}$$

Notice the initial downrange position y_0 only appears as an additive constant in Eq 2.46 and no where else. This will be important in later discussion. Finally, the state transition matrix can be formed [9:page 6].

$$\Phi = \begin{bmatrix} 4-3\cos\psi & 0 & 0 & \frac{\sin\psi}{\eta} & \frac{2}{\eta}(1-\cos\psi) & 0 \\ 6(\sin\psi-\psi) & 1 & 0 & \frac{3}{\eta}(\cos\psi-1) & \frac{4}{\eta}\sin\psi - \frac{3\psi}{\eta} & 0 \\ 0 & 0 & \cos\psi & 0 & 0 & \frac{\sin\psi}{\eta} \\ 3\eta\sin\psi & 0 & 0 & \cos\psi & 2\sin\psi & 0 \\ 6\eta(\cos\psi-1) & 0 & 0 & -2\sin\psi & -3+4\cos\psi & 0 \\ 0 & 0 & -\eta\sin\psi & 0 & 0 & \cos\psi \end{bmatrix} \tag{2.51}$$

where ψ is $\eta\delta t$ and δt is the sample time. Using this Φ matrix, the state propagation equation for satellite one can be written as

$$\hat{\mathbf{x}}_1(-) = \Phi_1\hat{\mathbf{x}}_1(+) \tag{2.52}$$

Generalizing this equation for s satellites gives

$$\hat{\mathbf{x}}(+) = \begin{bmatrix} \Phi_1 & 0 & \cdots & 0 \\ 0 & \Phi_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \Phi_s \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_1(-) \\ \hat{\mathbf{x}}_2(-) \\ \vdots \\ \hat{\mathbf{x}}_s(-) \end{bmatrix} = \Phi\hat{\mathbf{x}}(-) \tag{2.53}$$

Note the Φ matrix is block diagonal.

One of the possible problems associated with the standard Kalman filter that has been illustrated thus far is numerical instability. The truth model was implemented on the computer in double precision resulting in numerical accuracy to sixteen significant figures. Since the filter was intended to operate on board a simulated satellite, it was constrained to single precision or eight significant figures. A method for analysing the stability of a given filter is through use of a *condition number* $k(\mathbf{A})$ [4:399]. The condition number of a matrix $\mathbf{A}$ is

$$k(\mathbf{A}) = \frac{\sigma_{max}}{\sigma_{min}} \tag{2.54}$$

where σ^2_{max} and σ^2_{min} are the maximum and minimum eigenvalues of $\mathbf{A}^T\mathbf{A}$. One of the indicators of instability is a covariance matrix with negative eigenvalues because, by definition, the covariance is positive semidefinite [4:page 90].

When this filter was actually implemented, the covariance matrix did in fact have negative eigenvalues. Upon investigation of $k(\mathbf{A})$, it was discovered that the minimum and maximum eigenvalues of $\mathbf{P}^T\mathbf{P}$ were on the order of 10^{-17} and 10^2 respectively. This resulted in a condition number on the order of 10^9. Studies show numerical difficulties will be experienced as the condition number approaches 10^N, where N is the number of significant digits. It is evident that, as described, this filter cannot be implemented in single precision.

2.3.2 Second Filter. One method of solving the numerical problems is using the 'U-D covariance factorization' version of the Kalman filter [4:pages 392-399]. This form of filter achieves twice the precision with the same wordlength [4:400]. This performance results from factoring the covariance into

$$\mathbf{P}(-) = \mathbf{U}(-)\mathbf{D}(-)\mathbf{U}^T(-) \tag{2.55}$$

$$\mathbf{P}(+) = \mathbf{U}(+)\mathbf{D}(+)\mathbf{U}^T(+) \tag{2.56}$$

where the U matrix is unitary upper triangular and the D matrix is diagonal. This method guarantees the positive semidefiniteness of the covariance and the numerical stability and accuracy of the filter.

The algorithm begins with the same initial conditions that were used in the Kalman filter. The first step is to reduce the n-by-n $\mathbf{P}_0$ (where n is the number of states, in this case $n = 6s$) into the factored form $\mathbf{UDU}^T$ through the following steps: First, for the n^{th} column

$$\begin{aligned} D_{nn} &= P_{nn} \\ U_{in} &= \begin{cases} 1 & i = n \\ P_{in}/D_{nn} & i = n-1, n-2, \ldots, 1 \end{cases} \end{aligned} \tag{2.57}$$

Then for the remaining columns, $j = n-1, n-2, \ldots, 1$

$$\begin{aligned} D_{jj} &= P_{jj} - \textstyle\sum_{k=j+1}^{n} D_{kk} U_{jk}^2 \\ U_{ij} &= \begin{cases} 0 & i > j \\ 1 & i = j \\ \left[P_{ij} - \sum_{k=j+1}^{n} D_{kk} U_{ik} U_{jk} \right] / D_{jj} & i = j-1, j-2, \ldots, 1 \end{cases} \end{aligned} \tag{2.58}$$

This procedure can still be used if the $\mathbf{Q}_d$ and $\mathbf{R}_f$ matrices are non-diagonal (to be discussed later).

Since the initial values for U and D have been determined, the state can be propagated to the first update time. Begin with setting $\mathbf{U}(+) = \mathbf{U}_0$ and $\mathbf{D}(+) = \mathbf{D}_0$. Then form the n-by-$2n$ matrix $\mathbf{Y}(-)$ as

$$\mathbf{Y}(-) = [\, \mathbf{\Phi U}(+) \,|\, \mathbf{G}_d \,] \tag{2.59}$$

where $\mathbf{\Phi}$ and $\mathbf{G}_d$ are as previously defined. Finally, form the $2n$-by-$2n$ matrix $\tilde{\mathbf{D}}(-)$ by

$$\tilde{\mathbf{D}}(-) = \begin{bmatrix} \mathbf{D}(+) & 0 \\ 0 & \mathbf{Q}_d \end{bmatrix} \tag{2.60}$$

The propagation begins by forming

$$\mathbf{Y}^T(-) = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \tag{2.61}$$

and then calculating the following relationships for $k = n, n-1, \ldots, 1$

$$\begin{aligned}
\mathbf{c}_k &= \tilde{\mathbf{D}}(-)\mathbf{a}_k \quad \left(c_{kj} = \tilde{D}_{jj}(-)a_{kj}, \quad j = 1, 2, \ldots, 2n\right) \\
D_{kk}(-) &= \mathbf{a}_k^T \mathbf{c}_k \\
\mathbf{d}_k &= \mathbf{c}_k / D_{kk}(-) \\
U_{jk}(-) &= \mathbf{a}_j^T \mathbf{d}_k \qquad j = 1, 2, \ldots, k-1 \\
\mathbf{a}_j &\leftarrow \mathbf{a}_j - U_{jk}(-)\mathbf{a}_k \quad j = 1, 2, \ldots, k-1
\end{aligned} \tag{2.62}$$

The final portion of the $\mathbf{U} - \mathbf{D}$ covariance factorization filter algorithm is the *scaler* measurement update. Let the first row of Eq 2.39 be the 1-by-n matrix $\mathbf{H}_1$ and let z_1 be the corresponding scaler measurement of Eq 2.12. Lastly, let R_1 be the corresponding element of the diagonal matrix $\mathbf{R}_f$; if $\mathbf{R}_f$ were not diagonal, a Cholesky decomposition [4:375] would need to be computed. This decomposition would re-define the filter $\mathbf{z}$ vector and the $\mathbf{H}$ matrix but would then give a diagonal $\mathbf{R}_f$ matrix for use in this filter. One can now complete the update by

$$\begin{aligned}
\mathbf{f} &= \mathbf{U}^T\mathbf{H}^T \\
v_j &= D_{jj}(-)f_j \quad j = 1, 2, \ldots, n \\
a_0 &= R
\end{aligned} \tag{2.63}$$

Then, for $k = 1, 2, \ldots, n$

$$\begin{aligned}
a_k &= a_{k-1} + f_k v_k \\
D_{kk}(+) &= D_{kk}(-)a_{k-1}/a_k \\
v_k &\leftarrow v_k \\
p_k &= -f_k/ak - 1 \\
U_{jk}(+) &= U_{jk}(-) + b_j p_k \quad j = 1, 2, \ldots, (k-1) \\
b_j &\leftarrow b_j + U_{jk}(-)v_k \quad j = 1, 2, \ldots, (k-1)
\end{aligned} \tag{2.64}$$

The filter gain is

$$\mathbf{K} = \frac{\mathbf{b}}{a_n} \tag{2.65}$$

Once the gain has been determined, $\hat{\mathbf{x}}(+)$ and $\mathbf{P}(+)$ can be calculated by using Eqs 2.35 and 2.36.

Upon testing, it was discovered that the filter covariance would not converge; therefore, the filter did not track the true state. The maximum eigenvalue of the covariance and the true error were growing without bound. Further investigation of the estimated state revealed the y components were responsible for the growth in the true error. As the state was propagated through several orbits the downrange position error increased to more than 1000 kilometers. Therefore, it was evident this component was responsible for the filter instability.

It was stated earlier in this section that the U-D factorization filter guaranteed stability. This statement only considered numerical stability, it said nothing about filter stability. In order to guarantee filter stability, the system must be observable. The filter performance indicates that the *downrange* (y) component of the state is unobservable. This fact seems obvious once the filter has revealed it; the downrange displacement of the satellite cluster from the reference point has absolutely no effect on the relative range measurements or on the relative movement inside the cluster. Mathematically, this is verified by Eqs 2.45 through eq:compzd. Remembering the note on the initial downrange position, it can be seen that if the entire cluster is moved downrange from the reference point (by a constant increase in y_0 for each satellite), neither the other components of the state nor the range between satellites will be affected. Since the measurements contain no information on absolute downrange position, it is impossible to estimate the absolute y displacement from the reference point.

2.3.3 Third Filter. This instability problem can be resolved with the U-D factorization algorithm but the filter states must be modified in order to eliminate

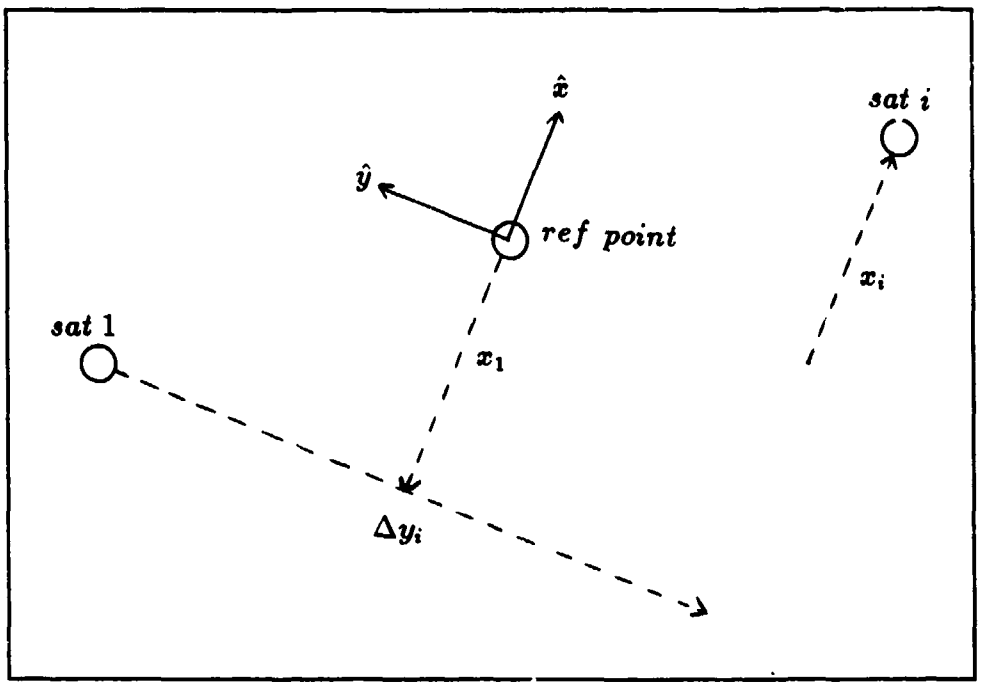

Figure 2.4. Modified State

unobservability. The modified state replaces the absolute downrange component with the relative downrange displacement Δy as illustrated in Figure 2.4. Where Δy is defined as

$$\Delta y_i = y_1 - y_i \tag{2.66}$$

For $i = 2, 3, \ldots, s$. Therefore, the new state vectors for $\mathbf{x}_1$ and $\mathbf{x}_i$ are:

$$\mathbf{x}_1 = \begin{bmatrix} x_1 \\ z_1 \\ \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{bmatrix} \quad and, \quad \mathbf{x}_i = \begin{bmatrix} x_i \\ \Delta y_i \\ z_i \\ \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \end{bmatrix} \tag{2.67}$$

Because the states have changed, the Φ matrix must also be modified. By examining Eqs 2.45 through 2.50, one sees that the only y, and therefore the only Δy, dependance is in Eq 2.46. Substituting Eq 2.46 into Eq 2.66 yields

$$\begin{aligned}\Delta y_i(+) &= [6(\sin\psi - \psi)](x_1 - x_i) + \Delta y_i + \\ &\left[\frac{2}{\eta}(\cos\psi - 1)\right](\dot{x}_1 - \dot{x}_i) + \left(\frac{4}{\eta}\sin\psi - \frac{3\psi}{\eta}\right)(\dot{y}_1 - \dot{y}_i) \qquad (2.68)\end{aligned}$$

where the state components on the right side of the equation are understood to be evaluated at the previous sample time. The effect of this modification on Φ_1 is to reduce it from a 6-by-6 to a 5-by-5 matrix by removing the row and column corresponding to the y component.

$$\Phi_1 = \begin{bmatrix} 4-3\cos\psi & 0 & \frac{\sin\psi}{\eta} & \frac{2}{\eta}(1-\cos\psi) & 0 \\ 0 & \cos\psi & 0 & 0 & \frac{\sin\psi}{\eta} \\ 3\eta\sin\psi & 0 & \cos\psi & 2\sin\psi & 0 \\ 6\eta(\cos\psi - 1) & 0 & -2\sin\psi & -3+4\cos\psi & 0 \\ 0 & -\eta\sin\psi & 0 & 0 & \cos\psi \end{bmatrix} \qquad (2.69)$$

The effect on Φ_i is that the second row of the matrix is now the negative of what it was in Eq 2.51 The second element (1) being the exception because of the way that Δy was defined in Eq 2.66.

$$\Phi_i = \begin{bmatrix} 4-3\cos\psi & 0 & 0 & \frac{\sin\psi}{\eta} & \frac{2}{\eta}(1-\cos\psi) & 0 \\ -6(\sin\psi - \psi) & 1 & 0 & -\frac{2}{\eta}(\cos\psi - 1) & -\frac{4}{\eta}\sin\psi + \frac{3\psi}{\eta} & 0 \\ 0 & 0 & \cos\psi & 0 & 0 & \frac{\sin\psi}{\eta} \\ 3\eta\sin\psi & 0 & 0 & \cos\psi & 2\sin\psi & 0 \\ 6\eta(\cos\psi - 1) & 0 & 0 & -2\sin\psi & -3+4\cos\psi & 0 \\ 0 & 0 & -\eta\sin\psi & 0 & 0 & \cos\psi \end{bmatrix} \qquad (2.70)$$

Unlike the structure of Eq 2.53, when the individual Φ_i matrices are combined into the composite state transition matrix, the result is *not* a block diagonal matrix. The non-diagonal elements are

$$\Phi_{1i} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 6(\sin\psi - \psi) & 0 & \frac{2}{\eta}(\cos\psi - 1) & \frac{4}{\eta}\sin\psi - \frac{3\psi}{\eta} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.71}$$

Thus, the overall system is of the form

$$\hat{\mathbf{x}}(-) = \begin{bmatrix} \Phi_1 & 0 & \cdots & 0 \\ \Phi_{12} & \Phi_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{1s} & 0 & 0 & \Phi_s \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_1(-) \\ \hat{\mathbf{x}}_2(-) \\ \vdots \\ \hat{\mathbf{x}}_s(-) \end{bmatrix} = \Phi\hat{\mathbf{x}}(+) \tag{2.72}$$

The measurement update portions of the filter ($\mathbf{h}$ and $\mathbf{H}$) also need some revision. The non-linear measurement vector h requires only a minor alteration. Substituting Eq 2.66 into Eq 2.37 gives

$$\mathbf{h} = \begin{bmatrix} \sqrt{(x_1 - x_2)^2 + \Delta y_2^2 + (z_1 - z_2)^2} \\ \vdots \\ \sqrt{(x_1 - x_s)^2 + \Delta y_s^2 + (z_1 - z_s)^2} \end{bmatrix} = \begin{bmatrix} h_1 \\ \vdots \\ h_{s-1} \end{bmatrix} \tag{2.73}$$

The linearization of this vector into the $\mathbf{H}$ matrix results in more extreme changes. Using a form similar to Eq 2.39, the new linearized matrix can be expressed as

$$\mathbf{H} = \begin{bmatrix} \tilde{H}_1 & H_1 & 0 & \cdots & 0 \\ \tilde{H}_2 & 0 & H_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{H}_{s-1} & 0 & 0 & \cdots & H_{s-1} \end{bmatrix} \tag{2.74}$$

where for $i = 1, 2, \ldots, s-1$

$$\tilde{H}_i = \begin{bmatrix} \frac{x_i - x_{i+1}}{h_i} & \frac{z_i - z_{i+1}}{h_i} & 0 & 0 & 0 \end{bmatrix} \tag{2.75}$$

$$H_i = \begin{bmatrix} -\left(\frac{x_i - x_{i+1}}{h_i}\right) & \frac{\Delta y_{i+1}}{h_i} & -\left(\frac{z_i - z_{i+1}}{h_i}\right) & 0 & 0 & 0 \end{bmatrix} \tag{2.76}$$

The modification is complete except for some final minor changes.

With the removal of the second diagonal terms of the $\mathbf{G}_d$, $\mathbf{P}_0$, and $\mathbf{Q}_d$ matrices, U-D covariance factorization algorithm is complete. The final step in building this filter is the tuning process during performance evaluations.

III. Performance Analysis

3.1 Analysis Tools

The truth model and the U-D factorization covariance filter have both been developed. All the necessary information for an initial performance analysis is available; specifically, the true state $\mathbf{x}_t$, the estimated state $\hat{\mathbf{x}}$, and the covariance $\mathbf{P}$ have all been computed. The remaining task is to convert this input data into useful output.

One of the tools for this data conversion is the true error $\mathbf{e}_t$ as depicted in Figure 2.1. The quantities of interest, $\hat{\mathbf{y}}$ and $\mathbf{y}_t$, can be determined from the state by

$$\mathbf{y}_t = \mathbf{C}_t \mathbf{x}_t \tag{3.1}$$

and

$$\hat{\mathbf{y}} = \mathbf{C}\hat{\mathbf{x}} \tag{3.2}$$

Since the components of the true state are the same as those of the estimated state,

$$\mathbf{C} = \mathbf{C}_t$$

$\mathbf{C}$ would be a 2-by-n matrix for satellite one and a 3-by-n for every other satellite (n equals the total number of states). For example,

$$\mathbf{y}_{t_1} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x_{t_1} \\ z_{t_1} \\ \vdots \\ \dot{z}_{t_1} \end{bmatrix} = \begin{bmatrix} x_{t_1} \\ z_{t_1} \end{bmatrix} \tag{3.3}$$

and the true error is given by Eq 2.1 as

$$\mathbf{e}_t = \hat{\mathbf{y}}_1 - \mathbf{y}_{t_1} = \begin{bmatrix} \hat{x}_1 - x_{t_1} \\ \hat{z}_1 - z_{t_1} \end{bmatrix} \tag{3.4}$$

The other tool needed for the analysis is the error covariance. If this system were a linear system driven by white Gaussian noise with zero mean processes, then the state covariance would also be the error covariance [4:page 335]. For the system in this study, the covariance can still be used to get a general feel for the filter performance by implementing a single sample Monte Carlo analysis. For a more accurate and complete analysis, a full Monte Carlo simulation would need to be accomplished.

In this scaled down study, the position components of the state are the variables of interest; therefore, the corresponding position portion of the error covariance $\mathbf{P}_e$ is needed and can be found by

$$\mathbf{P}_{e_i} = \mathbf{C}_i \mathbf{P} \mathbf{C}_i^T \tag{3.5}$$

where $\mathbf{C}_i$ is the same matrix used in Eqs 3.1 and 3.2, and where i designates the satellite of interest.

To simplify the analysis even further, the magnitude of the true error $|\mathbf{e}_{t_i}|$ is plotted against the standard deviation (σ_i) of the covariance. The standard deviation is found by taking the square root of the largest eigenvalue of $\mathbf{P}_{e_i}$. These approximations are extremely conservative and, therefore, do not negate the validity of the performance evaluation.

3.2 Sample Runs

Sample performance runs were made with varying numbers of satellites in the cluster and with differing values of $\mathbf{Q}_d$ and $\mathbf{P}_0$. All filter tuning was accomplished by adjusting $\mathbf{Q}_d$; the filter measurement covariance matrix $\mathbf{R}_f$ was equal to $\mathbf{R}_t$ for all time and was based on assumed range measurement errors of .01 meters. The

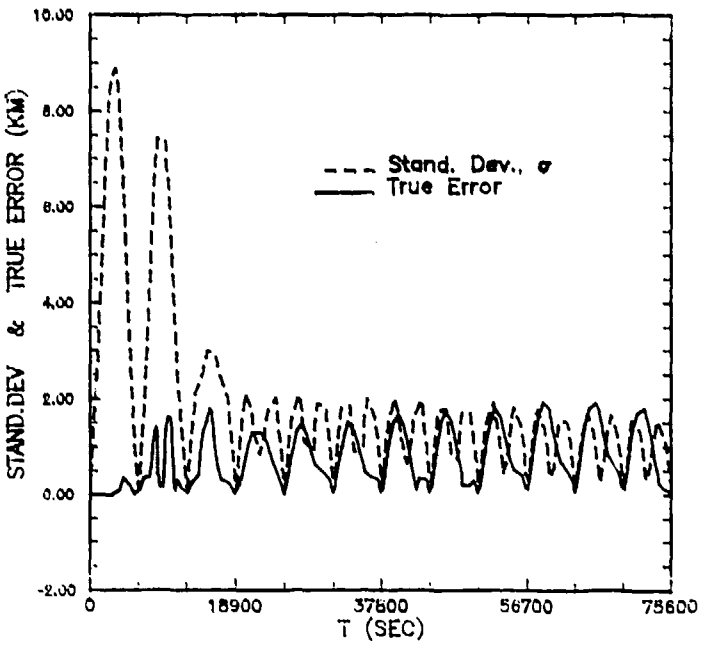

Figure 3.1. Filter performance (satellite 1 of 2, no dynamics noise).

cluster radius was 500 meters at a reference altitude of 1000 kilometers above the earth. This resulted in a cluster orbital period of approximately 6300 seconds.

The purpose of the first test run was to determine if the filter was stable. The system was initialized to two satellites with the estimator knowing true position and velocity. The dynamics driving noise ($\mathbf{Q}_d$) was turned off and the filter was fed perfect range data. Figure 3.1 shows the results. As hoped, the covariance converged thereby demonstrating filter stability. The divergence of the true error occurred as expected. Since the filter has been told to have perfect faith in its internal dynamics model ($\mathbf{Q}_d = 0$), the gain will converge to zero and decrease the effect of the range measurement inputs on the updated state. Also, note the true error and standard deviation are periodic with the orbit.

Since the basic stability of the filter was confirmed, the next test was to check for filter performance improvements with the addition of dynamics driving noise. The order of magnitude of the $\mathbf{Q}_d$ matrix diagonals was determined by calculating

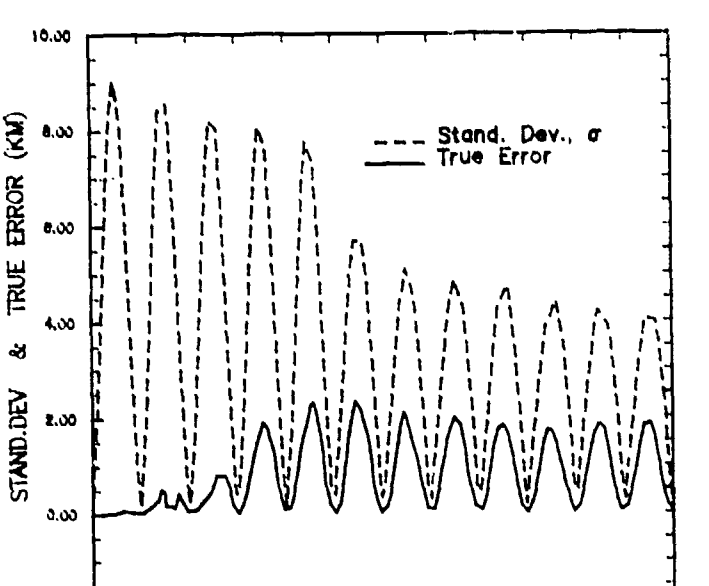

Figure 3.2. Filter performance (satellite 1 of 2, with dynamics noise).

the true error after the initial time propagation. The values used were 10^{-12} for position and 10^{-17} for velocity. The resulting plot (Figure 3.2) reflects a stable, well behaved filter. However, the Q_d values used were apparently too large. The desired end result of tuning is to have agreement between the filter's internally generated error estimates and the true error. In this case, the filter is overestimating the errors it is committing and putting too much weight on the measurements [4:page 338]. Another problem is the magnitude of the true error; two kilometers is well outside the allowable error tolerances outlined in Appendix A. Further tuning is required.

Another method for decreasing the true error is by increasing the cluster size. Each additional satellite provides more information to the state estimator consequently improving the position estimate of satellite one. Figure 3.3 shows the dramatic results of increasing the array to five elements. The initial transient is much smaller as are the σ and e_t values. However, the true performance is revealed when the scale size is expanded. Figure 3.4 depicts a filter significantly

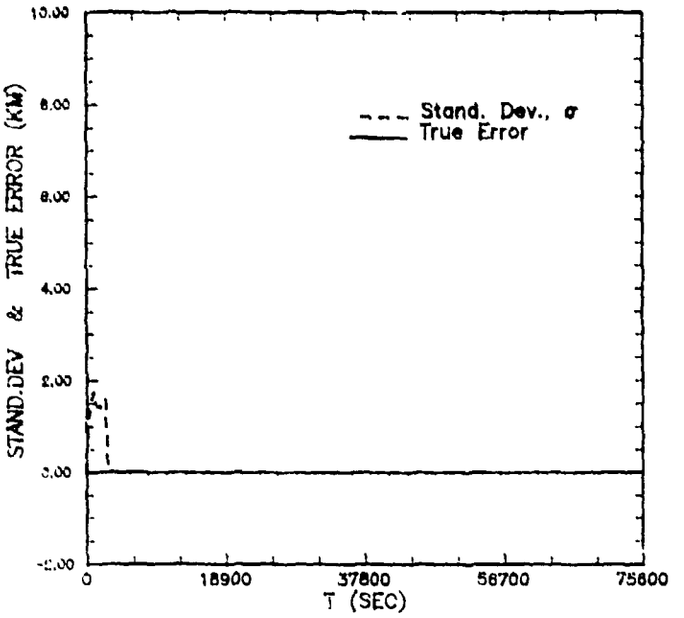

Figure 3.3. Filter performance (satellite 1 of 5, with dynamics noise).

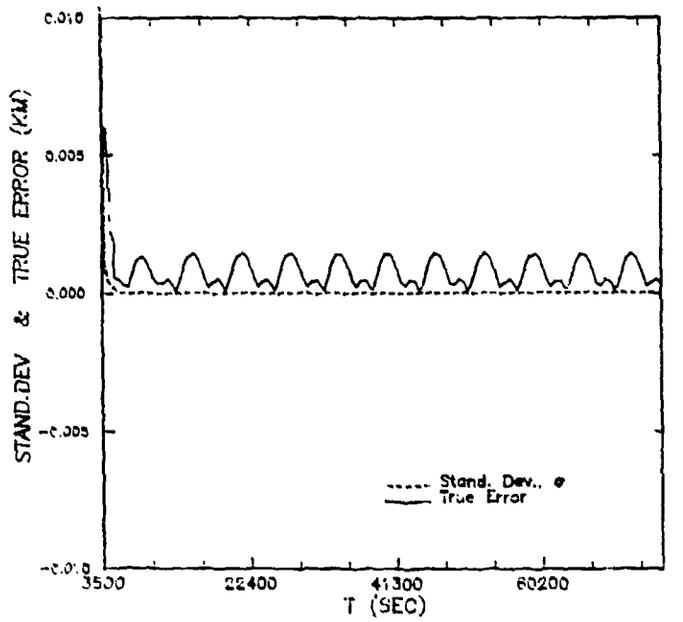

Figure 3.4. Filter performance (Figure 3.3, expanded).

underestimating its own error to the point of ignoring the measurement data. Even though more diagonal terms were added to the dynamics noise covariance with each additional satellite, it still was not enough to keep the filter gain from going to zero. The probable cause of this effect is significant unmodeled cross-correlation effects in the Q_d matrix. The time scale on the latest figure begins at 3500 seconds because the transient response of the filter was off the scale of the plot. Since the filter appears to be in steady-state within three orbital periods, future plots will terminate at 18900 seconds.

The objectives for the next filter test were: to reduce the transient response; to increase the cluster to its maximum size of ten satellites; and to do a rough retuning in order to evaluate performance. The transient response was improved by replacing the diagonal elements of the P_0 matrix with those of the steady-state covariance matrix. The new diagonals were on the order of 10^{-6} for position and 10^{-12} for velocity. Also, the dynamics noise covariance was increased to 2×10^{-9} and 2×10^{-14} for position and velocity variances respectively. Figures 3.5 and 3.6 show the results of these changes. The ten satellite filter performed exceptionally well. The transient response of the standard deviation was reduced from a peak of approximately 3 kilometers to a maximum of around 2.5 meters (Figure 3.6). The maximum true error was one tenth of a meter for satellite one and close to zero for the others. It seemed reasonable to expect satellite one's position to be known best because all the range measurements to every cluster element included information on this single satellite. The only apparent explanation lies in the differences between the state model of satellite one and the model of the other satellite states as seen in Eq 2.67.

On this run, all the filter statistics remain the same as in the previous test. The only change is that now *corrupted* data will be fed into the estimator. In the previous case, with perfect measurements, the true error was less than one tenth of a meter. The true error response to range data corrupted at .01m is

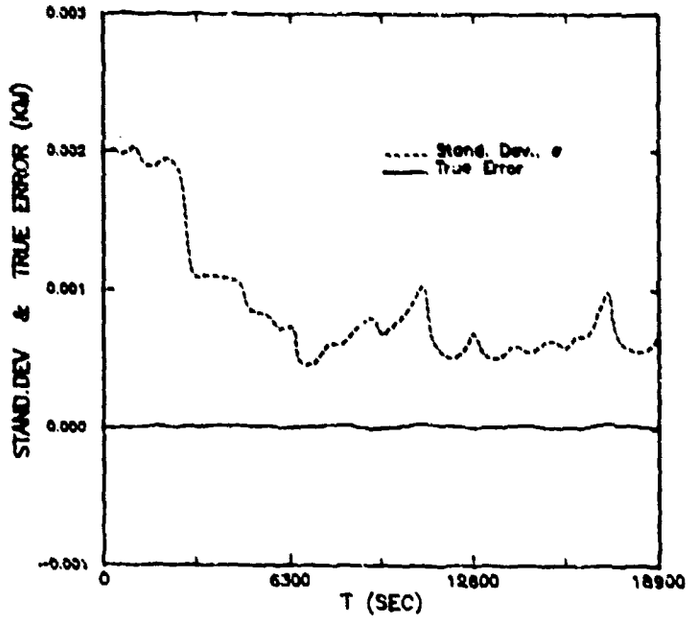

Figure 3.5. Filter performance (satellite 1 of 10, with dynamics noise and steady-state covariance).

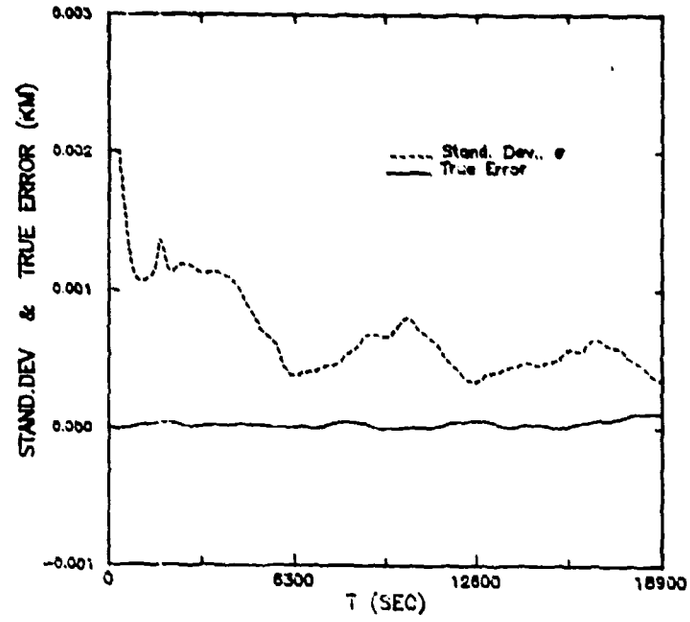

Figure 3.6. Filter performance (satellite 10 of 10, with dynamics noise and steady-state covariance).

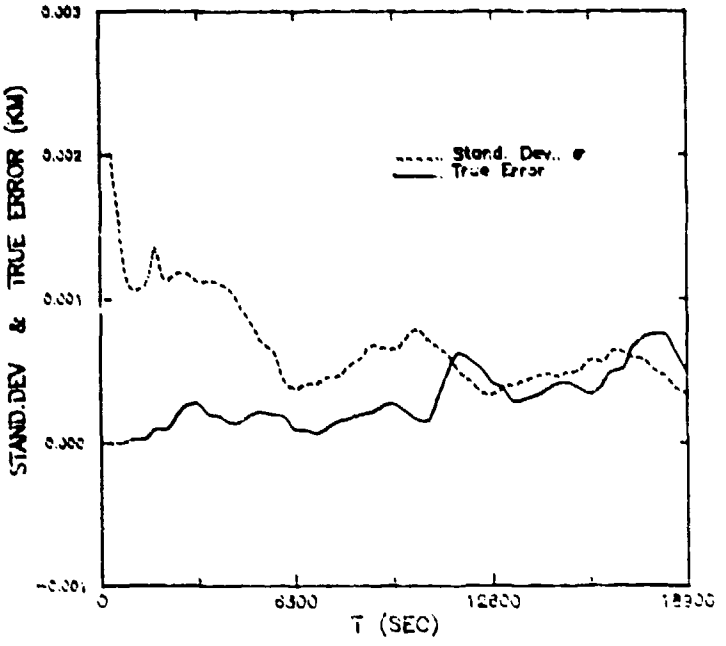

Figure 3.7. Filter performance (satellite 1 of 10, with dynamics noise, steady-state covariance, and noisy data).

seen in Figures 3.7, 3.8, and 3.9. The results were in agreement with what was anticipated. The true errors grew larger. In the case of satellite number one, the error grew to the point that it exceeded the internally estimated error. But this could be corrected by again retuning the filter. Even without retuning, the total error was still less than one meter which is well within the minimum required accuracy.

Finally, as a follow on to the previous test run, the filter was retuned to improve performance. The dynamics noise covariance was increased to 8×10^{-9} and 8×10^{14} for position and velocity variances respectively. Figures 3.10, 3.11, and 3.12 show that the performance was improved by minimal tuning, as claimed.

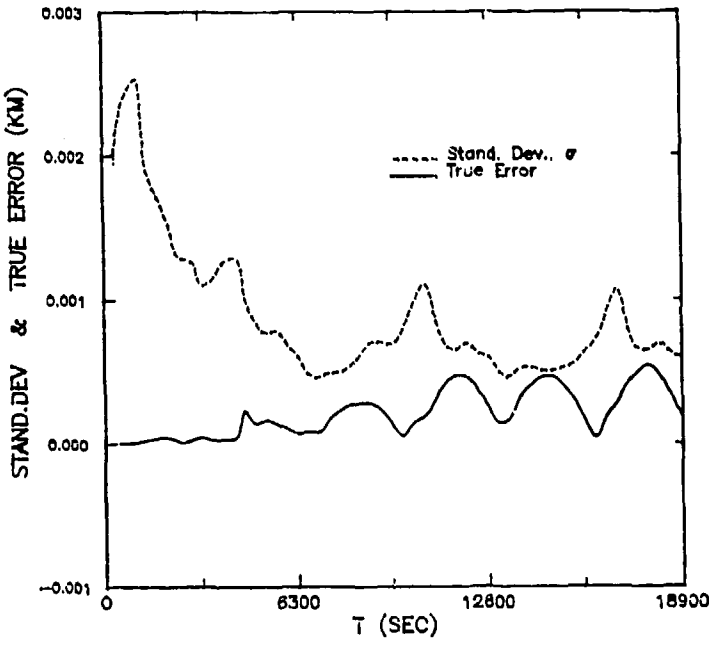

Figure 3.8. Filter performance (satellite 5 of 10, with dynamics noise, steady-state covariance, and noisy data).

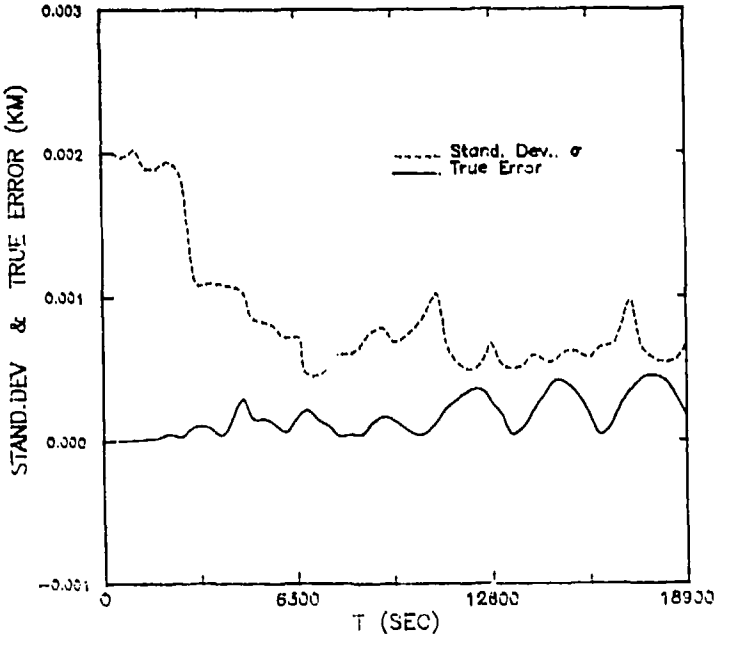

Figure 3.9. Filter performance (satellite 10 of 10, with dynamics noise, steady-state covariance, and noisy data).

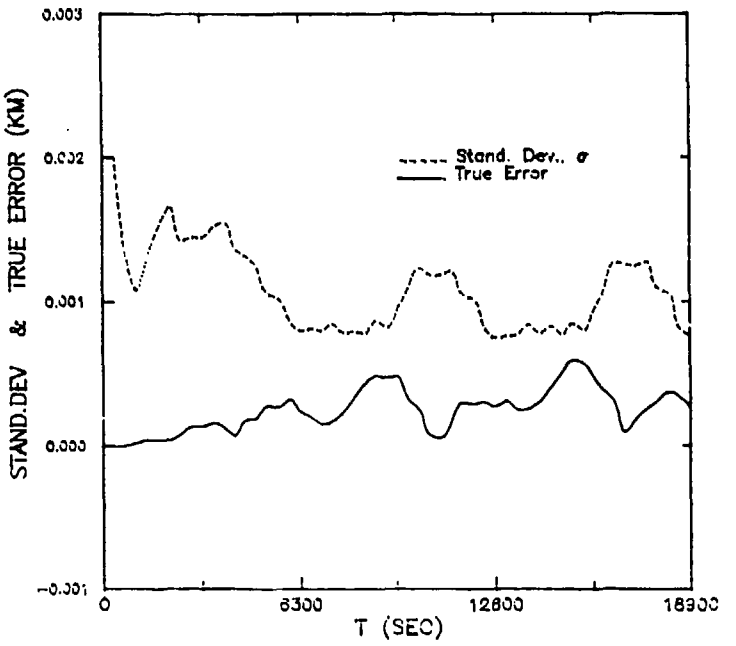

Figure 3.10. Filter performance (satellite 1 of 10, retuned, with noisy data).

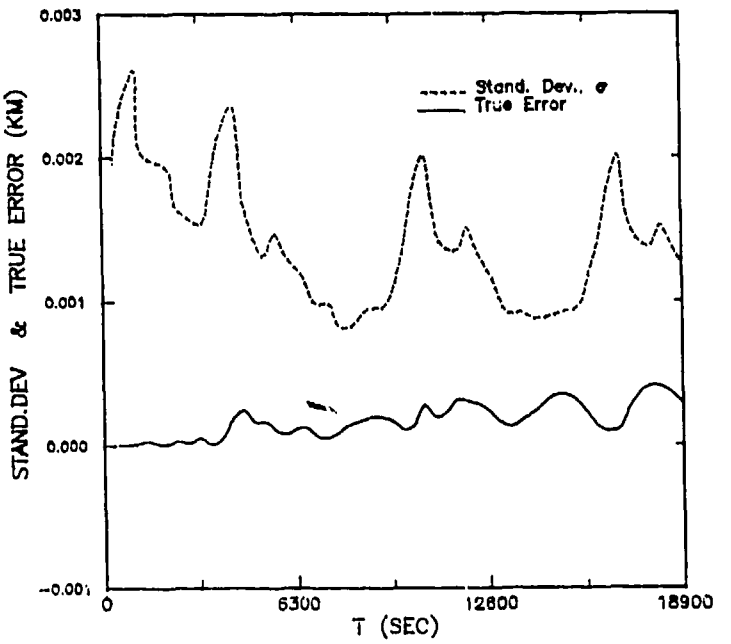

Figure 3.11. Filter performance (satellite 5 of 10, retuned, with noisy data).

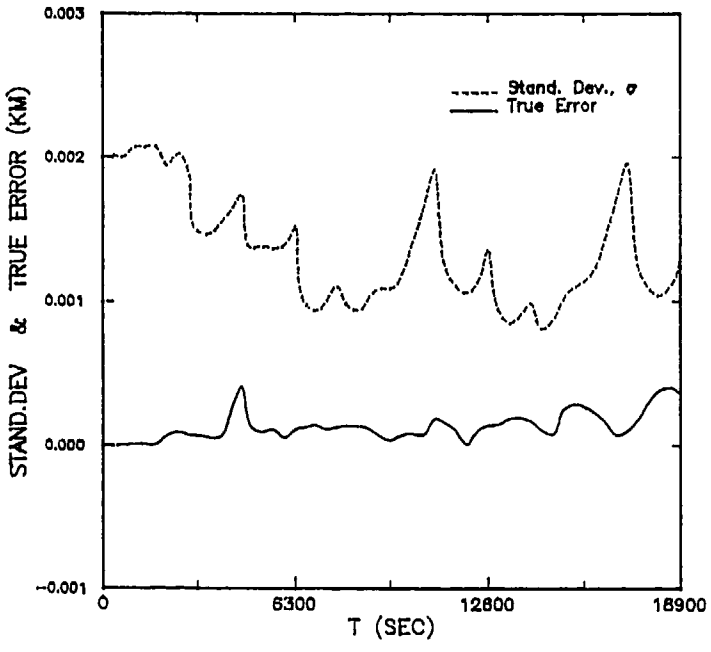

Figure 3.12. Filter performance (satellite 10 of 10, retuned, with noisy data).

IV. Conclusion

The U-D factorization filter performed well for all cases evaluated. Even though tuning was limited to the dynamics noise matrix, the estimator was able to significantly exceed the minimum required accuracy. Numerical stability and filter stability were maintained during all runs and the initial transients were successfully reduced by many orders of magnitude. All of this was accomplished with a cluster size significantly larger than previously tested. The analysis performed in this thesis was only an initial evaluation there remains much still to do.

A full Monte Carlo analysis could be conducted to determine the actual filter error statistics including biases. Minimize these errors through tuning. In the tuning process, investigate the performance enhancement in adjusting non-diagonal terms in the $\mathbf{Q}_d$ matrix. Do not limit the $\mathbf{R}_f$ matrix to the values of the truth model.

Test the robustness of the filter. Analyze how accurate the initial conditions on the state must be for the estimator to lock on. Test how noisy the range data can be and still obtain valid estimates. How far can the cluster be dispersed before the assumptions in the Clohessy-Wiltshire equations become invalid and nullify the mathematical basis of the filter? Finally, test the filter against a more accurate, *real word* truth model.

Appendix A. *Satellite Precision Requirement*

A.1 Introduction

This appendix outlines how the position accuracy requirements were determined. These requirements were used to evaluate filter performance.

A.2 Accuracy Requirement

As stated in the main introduction, the positions of the satellites must be known to at least one quarter of the radar wavelength. The maximum wavelength is a function of the size of the antenna.

According to the problem description, the cluster consists of ten satellites at an altitude of 1000 km. Assume that each satellite has a ground coverage area approximating that of Wyoming. Estimate the radius of this area as 480 km. Therefore, the angle subtended at the satellite by the ground is

$$\tan\left(\frac{\theta}{2}\right) = \frac{480}{1000} \tag{A.1}$$

defining this angle as the 3-dB beamwidth and solving for it gives

$$\theta_{3dB} = 2\tan^{-1}\left(\frac{480}{1000}\right) \tag{A.2}$$

Assuming that the antenna efficiency ξ is 55 percent, the antenna gain is found by [7:page 81]

$$G \simeq \frac{30000}{(\theta_{3dB})^2} \tag{A.3}$$

If the conservative estimate is made that the overall gain is just the multiple of the individual gains then

$$G_{TOT} = 10 \times G \tag{A.4}$$

The radar wavelength can then be found by

$$G_{TOT} = \xi\frac{4\pi A}{\lambda^2} \tag{A.5}$$

where A is the cluster area. Finally, solving for λ

$$\lambda = \pi r \left(\frac{\xi}{G_{TOT}} \right)^{1/2} \tag{A.6}$$

where r is the cluster radius. Remember, the needed accuracy is at least twenty-five percent of this wavelength. Table A.1 gives some representative values of minimum required filter performance.

cluster radius	$\lambda/4$
.5 km	2.75 m
5 km	275 m
50 km	2750 m

Table A.1. Cluster radius vs required accuracy

Bibliography

1. Bate, Roger R. and others. Fundamentals of Astrodynamics. New York: Dover Publications, 1971.
2. Bellaccini, S. and others. "Synchronization and Routing Aspects in a Cluster of Satellites with On-Board Processing," IEEE Global Telecommunications Conference 1986. 1713-1719. New York: IEEE Press,1986.
3. Jacobs, Donald H. Fundamentals of Optical Engineering. New York: McGraw-Hill Book Company, 1943.
4. Maybeck, Peter S. Stochastic Models, Estimation, and Control Vol. I. New York: Academic Press, 1982.
5. Maybeck, Peter S. Stochastic Models, Estimation, and Control Vol. II. New York: Academic Press, 1982.
6. Murdoch, J. and J. J. Pocha. "The Orbit Dynamics of Satellite Clusters," Space 2000, Selection of Papers Presented at the 33rd Congress of the International Astronautical Federation. 164-188. New York: American Institute of Aeronautics and Astronautics, Inc., 1983.
7. Pratt, Timothy and Charles W. Bostian. Satellite Communications. New York: John Wiley & Sons, 1986.
8. Swinerd, Graham G. and John Murdoch. "An Assessment of Satellite-to-Satellite Tracking Applied to Satellite Clusters," Proceedings of the Astrodynamics 1985 Conference. 1001-1018. San Diego: Univelt, Inc., 1986.
9. Wiesel, William B. Class handout distributed in MECH 532, Fundamentals of Astrodynamics. Air Force Institute of Technology (AU), Wright-Paterson AFB OH, September 1987.

Vita

Captain Michael L. P. Ward was born on 13 October 1958 in Roswell, New Mexico. He graduated from Cooper Senior High School in Abilene, Texas, in 1976. Later, he attended the Unites States Air Force Academy, from which he received the degree of Bachelor of Science in Astronautical Engineering in May 1980. Upon graduation, he received a USAF commission and traveled to Williams AFB, Arizona for pilot training. After earning his wings, he served as a C-130 pilot in the 50th Tactical Airlift Squadron and as a flight instructor and evaluator in the 16th Tactical Airlift Training Squadron, Little Rock AFB, Arkansas. In May 1987, he entered the School of Engineering at the Air Force Institute of Technology.

Permanent address: 324 Woodhill Rd
Newtown,Pennsylvania
18940

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GA/AA/88D-13	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION School of Engineering 6b. OFFICE SYMBOL (If applicable) AFIT/ENY	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology (AU) Wright-Patterson AFB OH 45433-6583	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
ESTIMATED SATELLITE CLUSTER ELEMENTS IN NEAR CIRCULAR ORBIT

12. PERSONAL AUTHOR(S)
Michael L. P. Ward, B.S., Capt, USAF

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
MS Thesis	FROM _____ TO _____	1988 December	49

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Satellite Cluster, Kalman Filter, Position Estimation
22	03		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Thesis Advisor: William E. Wiesel
Professor
Department of Aeronaautics and Astronautics

[signature] 12 Jan 1989

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
William E. Wiesel, Professor	(513) 255-3069	AFIT/ENY

DD Form 1473, JUN 86 — Previous editions are obsolete. — SECURITY CLASSIFICATION OF THIS PAGE

The relative position determination of the satellites in an orbital cluster is investigated. The on-board estimator is the U-D covariance factorization filter with dynamics based on the Clohessy-Wiltshire equations. Performance is measured by comparing the square root of the position covariance eigenvalues to the magnitude of true position errors. True errors are also compared to minimum accuracy requirements. Filter tuning is limited to adjusting the diagonal entries of the dynamics noise matrix. Test cases include: perfect initial conditions with perfect range data; and perfect initial conditions with noise corrupted measurements.